

FIG. 1

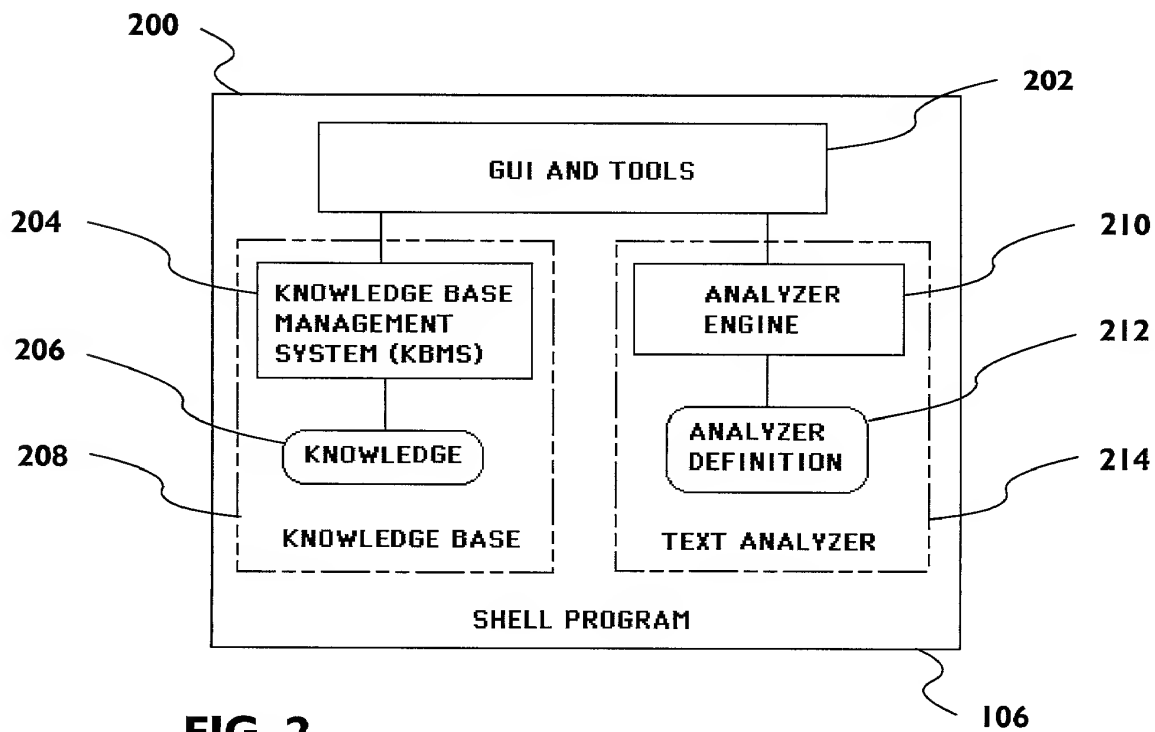


FIG. 2

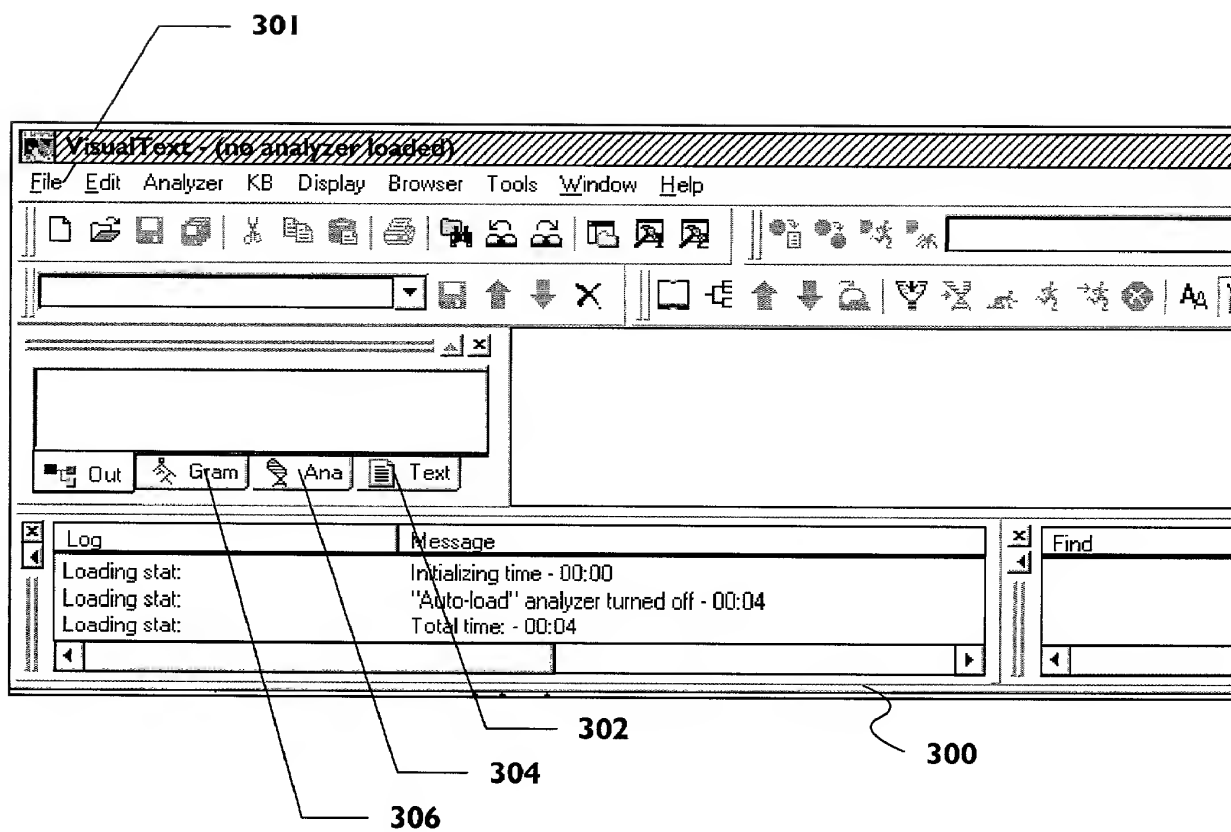


FIG. 3

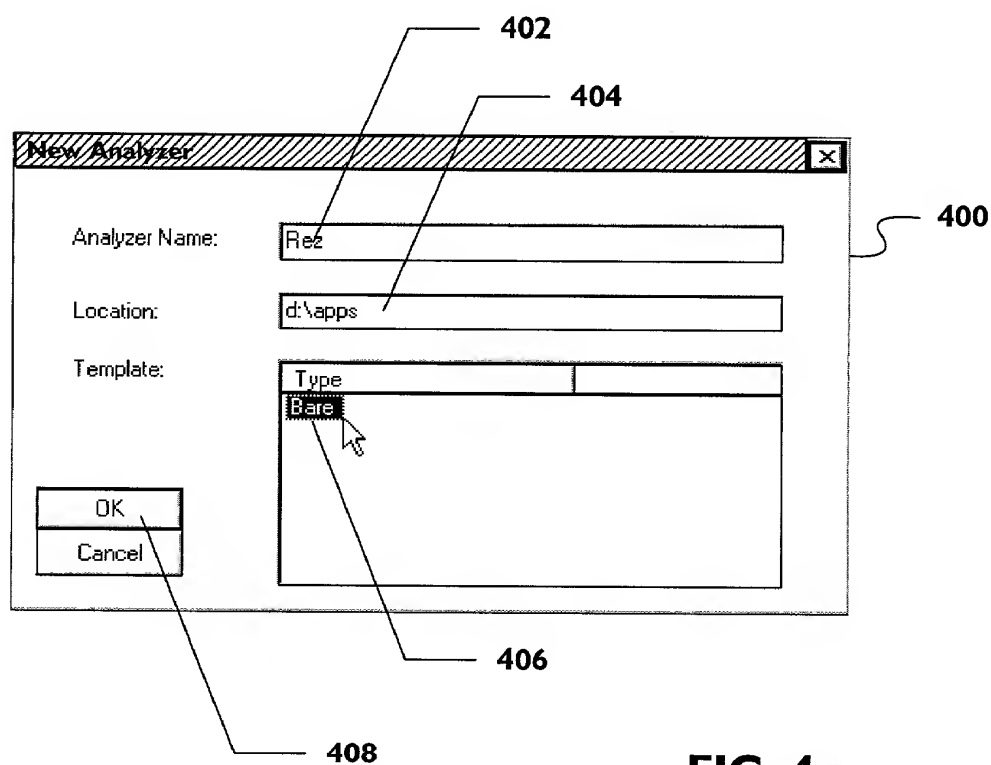


FIG. 4

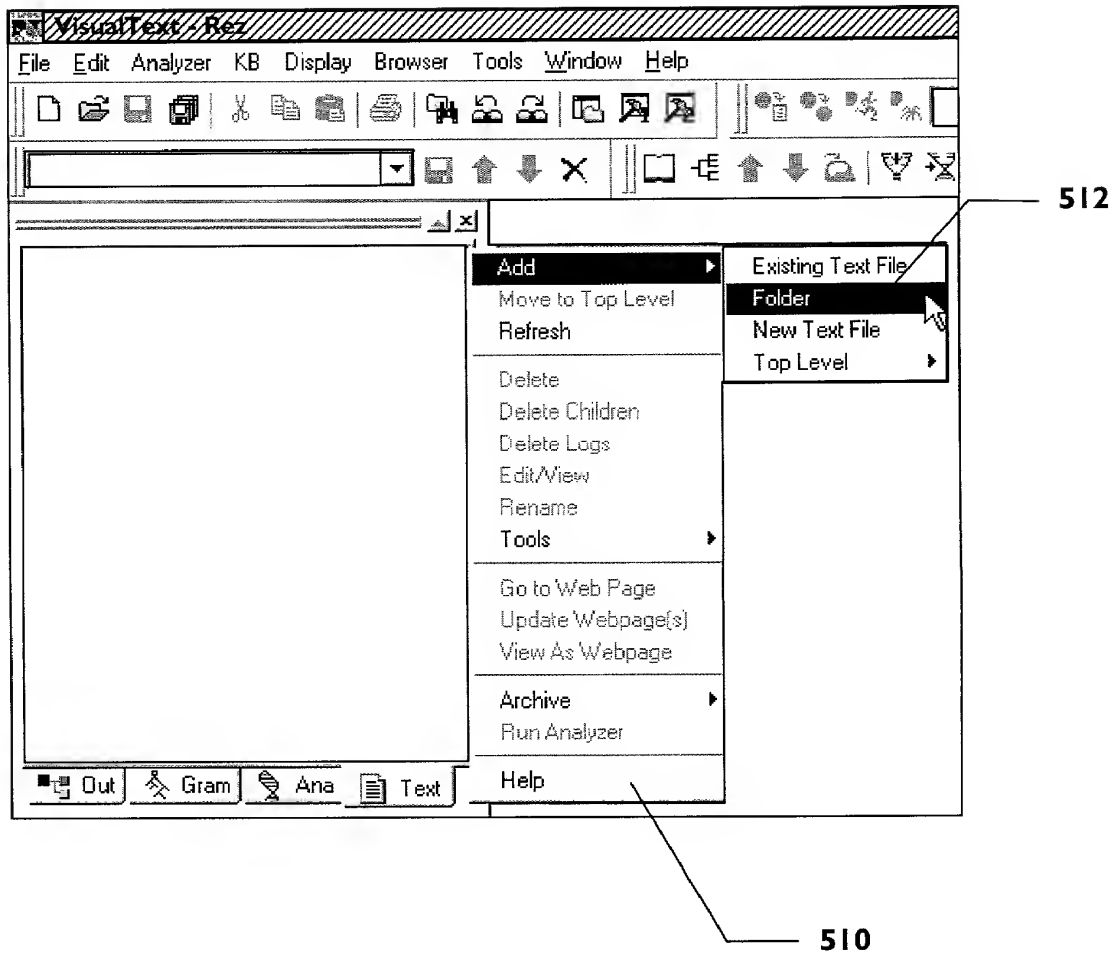


FIG. 5

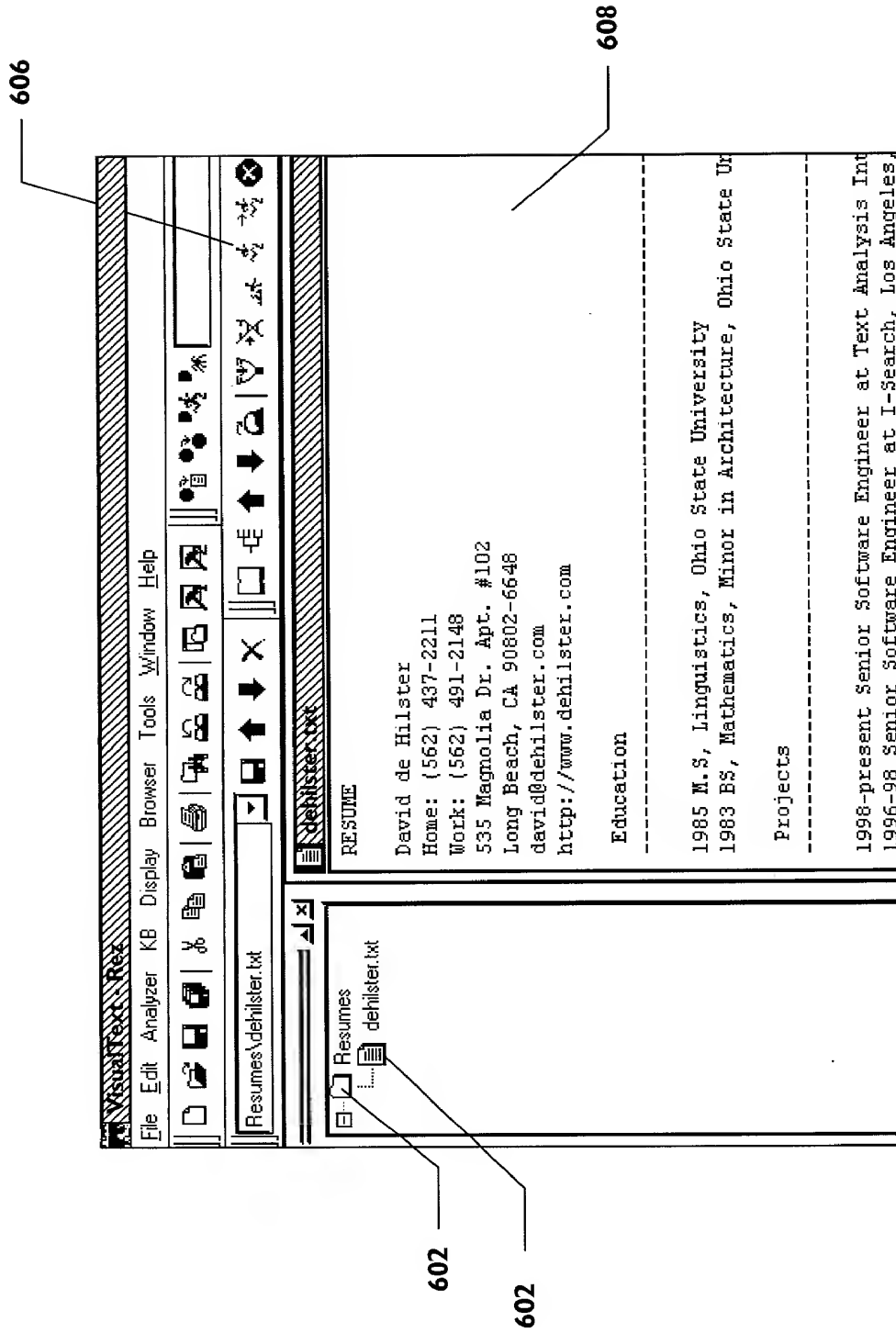


FIG. 6

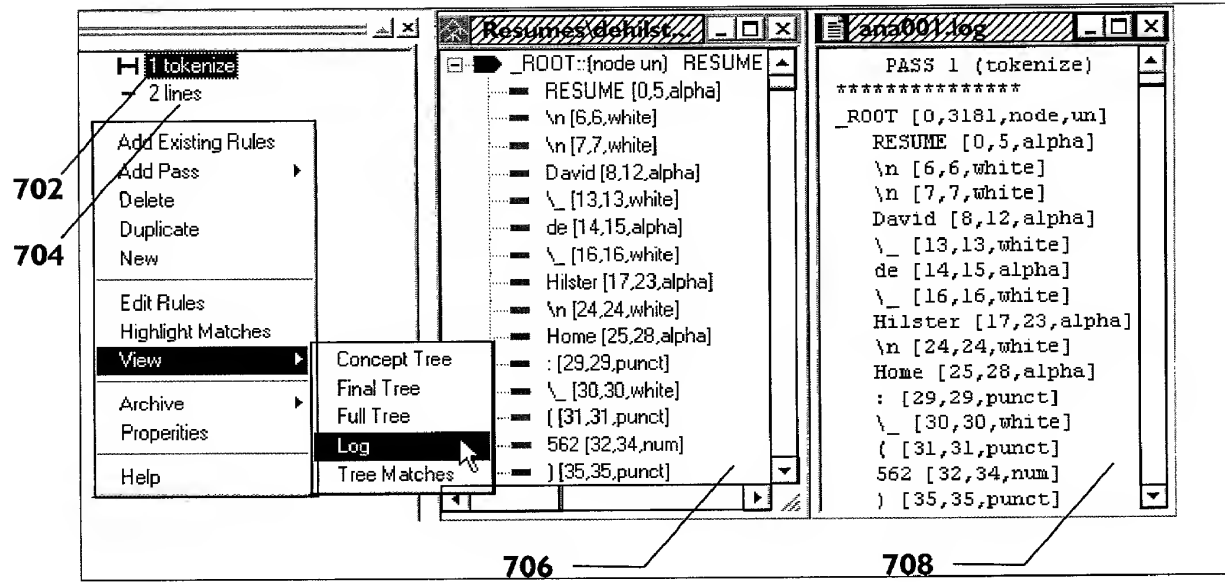


FIG. 7

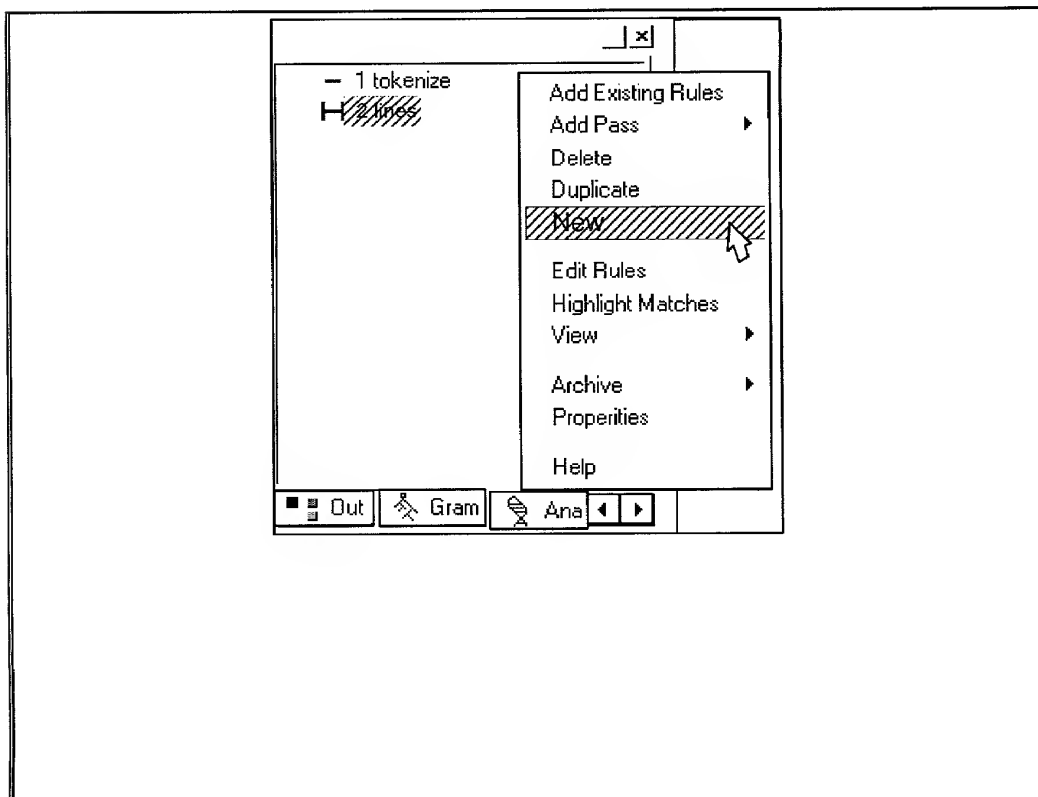


FIG. 8

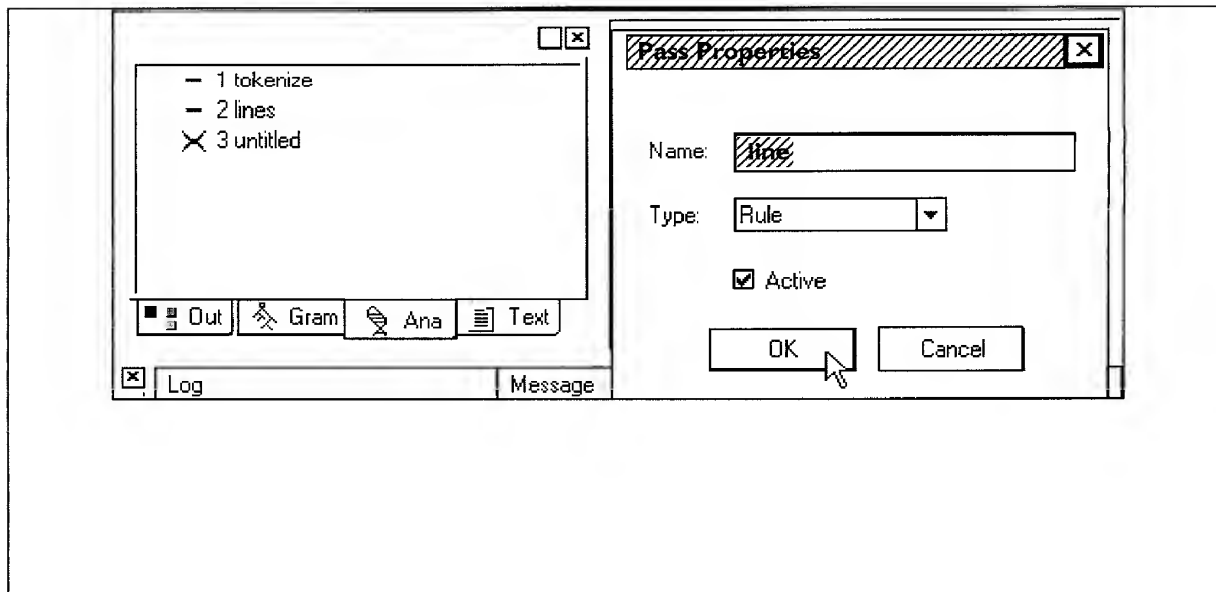


FIG. 9

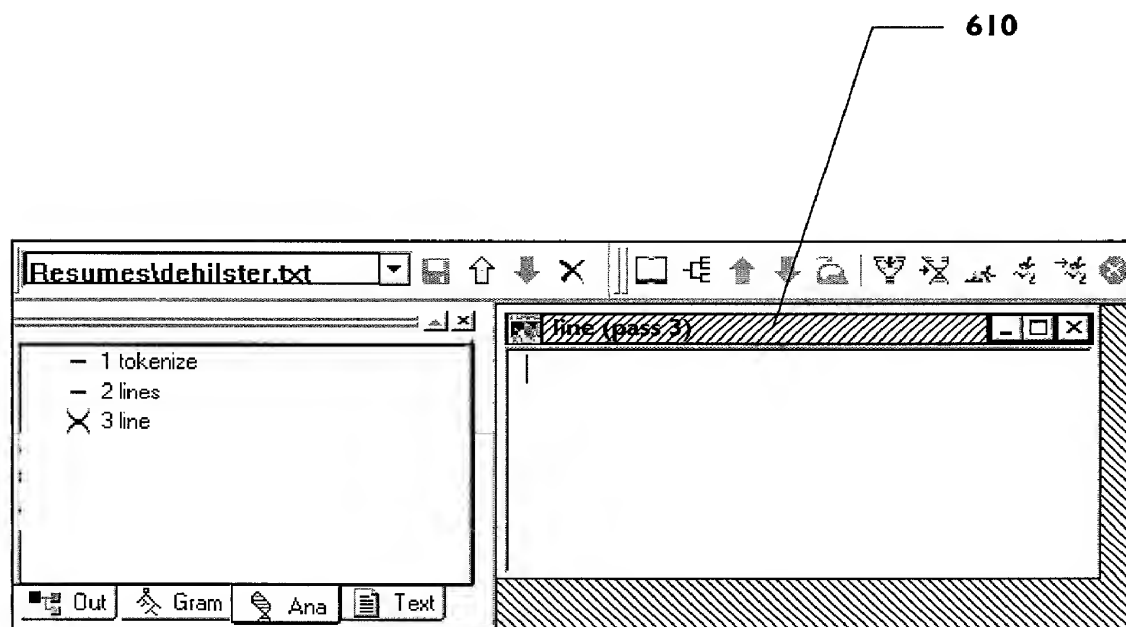
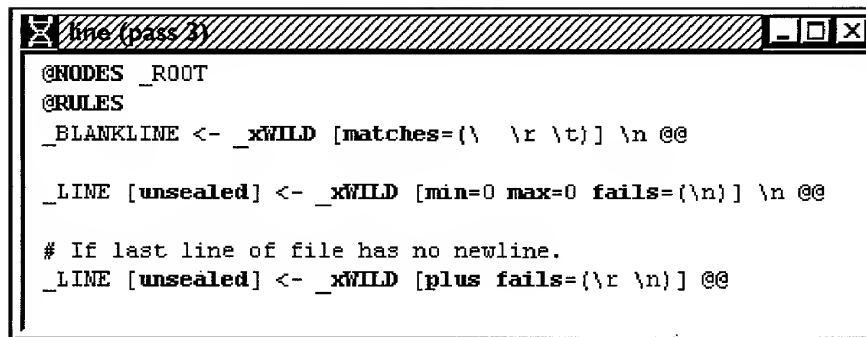


FIG. 10



```

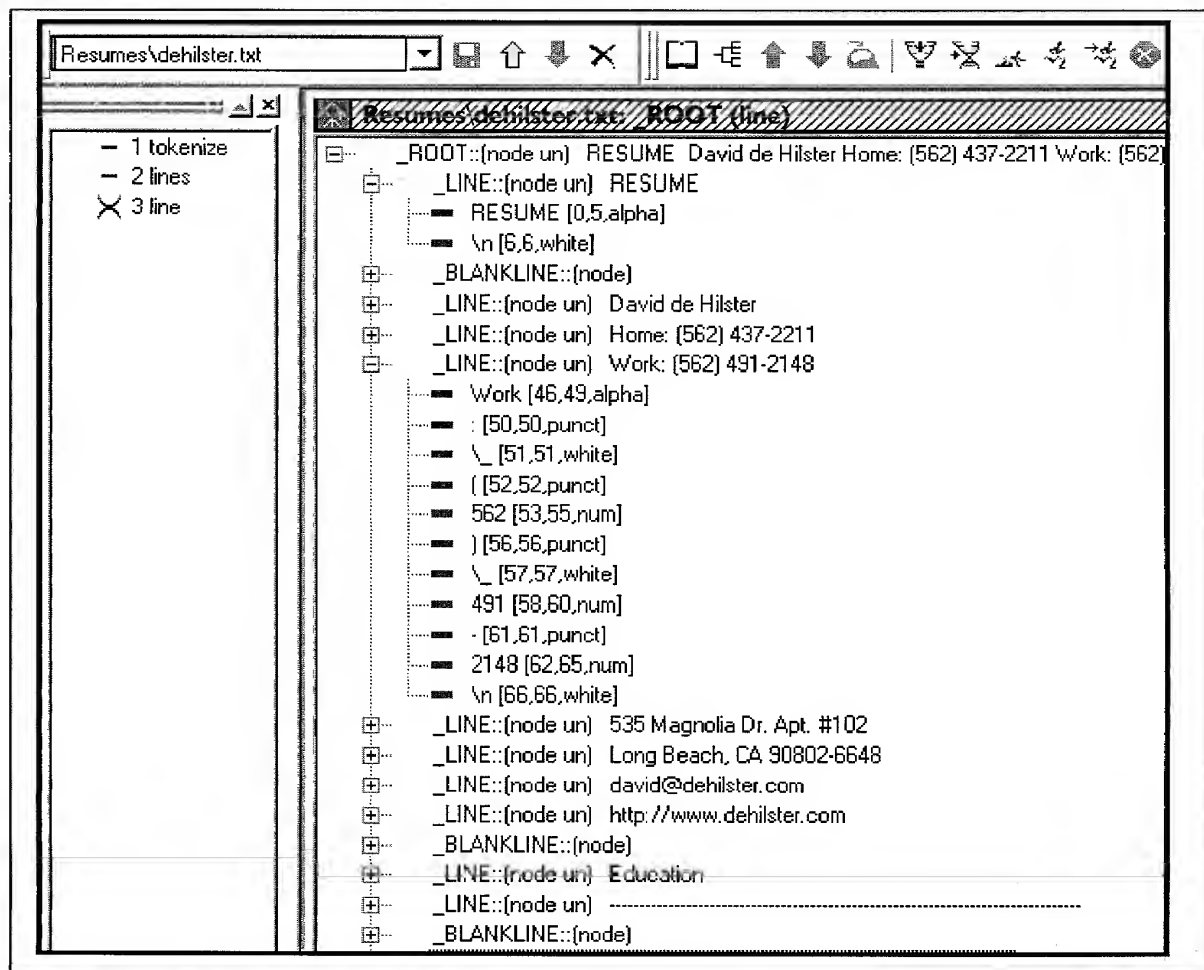
@NODES _ROOT
@RULES
 _BLANKLINE <- _xWILD [matches=(\  \r \t)] \n @@

 _LINE [unsealed] <- _xWILD [min=0 max=0 fails=(\n)] \n @@

# If last line of file has no newline.
 _LINE [unsealed] <- _xWILD [plus fails=(\r \n)] @@

```

FIG. 11



Resumes\dehilster.txt

1 tokenize  
2 lines  
3 line

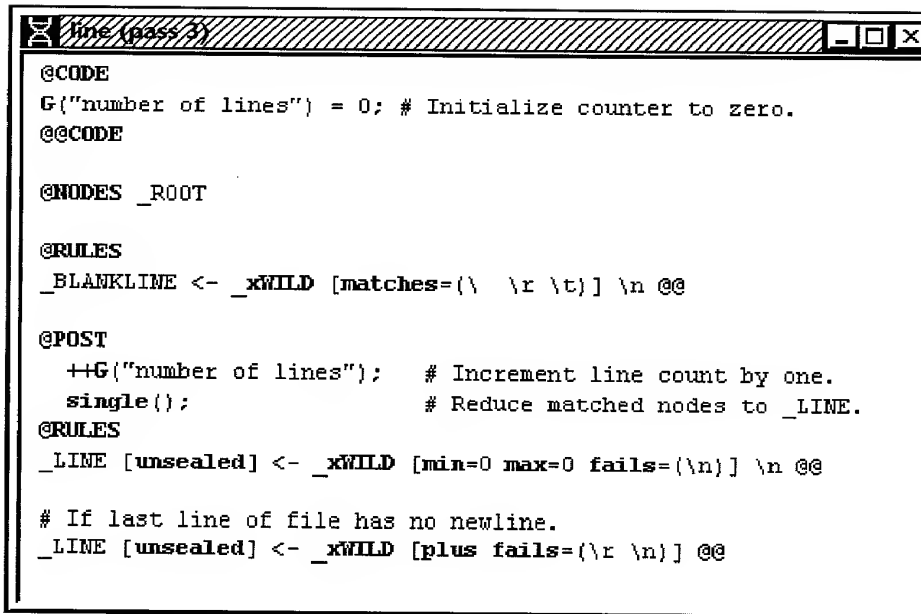
Resumes\dehilster.txt: ROOT (line)

```

_ROOT::(node un) RESUME David de Hilster Home: (562) 437-2211 Work: (562)
├── _LINE::(node un) RESUME
│   ├── RESUME [0,5,alpha]
│   └── \n [6,6,white]
├── _BLANKLINE::(node)
├── _LINE::(node un) David de Hilster
├── _LINE::(node un) Home: (562) 437-2211
├── _LINE::(node un) Work: (562) 491-2148
│   ├── Work [46,49,alpha]
│   ├── : [50,50,punct]
│   ├── \_ [51,51,white]
│   ├── ( [52,52,punct]
│   ├── 562 [53,55,num]
│   ├── ) [56,56,punct]
│   ├── \_ [57,57,white]
│   ├── 491 [58,60,num]
│   ├── - [61,61,punct]
│   ├── 2148 [62,65,num]
│   └── \n [66,66,white]
├── _LINE::(node un) 535 Magnolia Dr. Apt. #102
├── _LINE::(node un) Long Beach, CA 90802-6648
├── _LINE::(node un) david@dehilster.com
├── _LINE::(node un) http://www.dehilster.com
├── _BLANKLINE::(node)
├── _LINE::(node un) Education
├── _LINE::(node un) -----
└── _BLANKLINE::(node)

```

FIG. 12



```

@CODE
G("number of lines") = 0; # Initialize counter to zero.
@@CODE

@NODES _ROOT

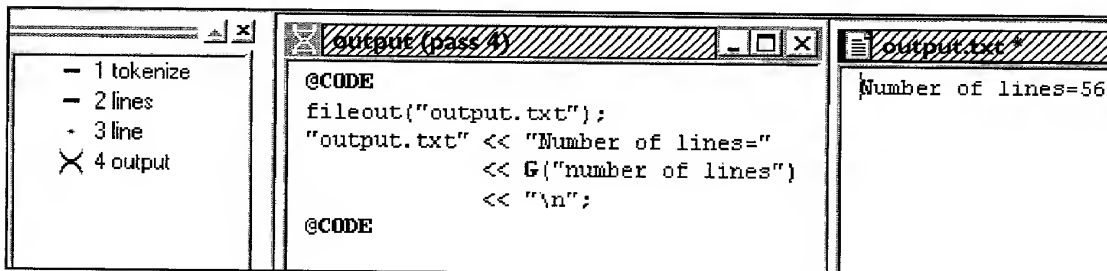
@RULES
_BLANKLINE <- _xWILD [matches=(\  \r \t)] \n @@

@POST
++G("number of lines"); # Increment line count by one.
single(); # Reduce matched nodes to _LINE.
@RULES
_LINE [unsealed] <- _xWILD [min=0 max=0 fails=(\n)] \n @@

# If last line of file has no newline.
_LINE [unsealed] <- _xWILD [plus fails=(\r \n)] @@

```

FIG. 13



- 1 tokenize
- 2 lines
- 3 line
- X 4 output

```

@CODE
fileout("output.txt");
"output.txt" << "Number of lines="
               << G("number of lines")
               << "\n";
@CODE

```

```

Number of lines=56

```

FIG. 14

```

@PATH _ROOT _educationZone _educationInstance _LINE

@POST
if (!X("city",3))
  X("city",3) = N("$text");
# noop()
@RULES
_xNIL <- _city [ ] @@

```

FIG. 15